L Number	Hits	Search Text	DB	Time stamp
-	40	chiffrement algorithme	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 14:46
-	19565	(encrypt$4 algorithm$2) with circuit	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 14:47
-	454	(encrypt$4 algorithm$2) with circuit with parallel	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 14:49
-	513	(encrypt$4 algorithm$2) with circuit with (parallel ("same time"))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 14:50
-	615	(encrypt$4 algorithm$2) with circuit with (parallel simultaneous$3 ("same time"))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 14:50
-	304	((encrypt$4 algorithm$2) with circuit with (parallel simultaneous$3 ("same time"))) same (throughput perform$4 output product$4 efficien$4)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 14:52
-	8	((encrypt$4 algorithm$2) with circuit with (parallel simultaneous$3 ("same time"))) same (throughput perform$4 output product$4 efficien$4) and (symmetry asymmetry)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 15:06
-	23	((encrypt$4 algorithm$2) with circuit with (parallel simultaneous$3 ("same time"))) same (throughput perform$4 output product$4 efficien$4) and (symmetr$2 asymmetr$2)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 15:30
-	51	((encrypt$4 algorithm$2) with (circuit appaatus device) with (parallel simultaneous$3 ("same time"))) same (throughput perform$4 output product$4 efficien$4) and (symmetr$2 asymmetr$2)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 15:06
-	54	((encrypt$4 algorithm$2) with (circuit apparatus device) with (parallel simultaneous$3 ("same time"))) same (throughput perform$4 output product$4 efficien$4) and (symmetr$2 asymmetr$2)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 15:07
-	29	((encrypt$4 algorithm$2) with (circuit apparatus device) with (parallel simultaneous$3 ("same time"))) with (throughput perform$4 output product$4 efficien$4) and (symmetr$2 asymmetr$2)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:28
-	25	(((encrypt$4 algorithm$2) with (circuit apparatus device) with (parallel simultaneous$3 ("same time"))) same (throughput perform$4 output product$4 efficien$4) and (symmetr$2 asymmetr$2)) not (((encrypt$4 algorithm$2) with (circuit apparatus device) with (parallel simultaneous$3 ("same time"))) with (throughput perform$4 output product$4 efficien$4) and (symmetr$2 asymmetr$2))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 15:23
-	67	((encrypt$4 algorithm$2) with circuit with (parallel simultaneous$3 ("same time"))) and ((data information) with (inaccessible separate isolat$4 access$3) with (system))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 15:33

-	10685	((encrypt$4 algorithm$2) with circuit with (parallel simultaneous$3 ("same time"))) and ((data information) with (inaccessible separate isolat$4 access$3) with (system)) ((double dual) adj port)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 15:34
-	6	((encrypt$4 algorithm$2) with circuit with (parallel simultaneous$3 ("same time"))) and ((data information) with (inaccessible separate isolat$4 access$3) with (system)) and ((double dual) adj port)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 15:41
-	12	((encrypt$4 algorithm$2) with circuit with (parallel simultaneous$3 ("same time"))) and (isolat$4 or separat$4) and ((double dual) adj (bus memory port))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 15:54
-	111	((encrypt$4 algorithm$2) with (hardware apparatus design system circuit) with (parallel simultaneous$3 ("same time"))) and (isolat$4 or separat$4) and ((double dual) adj (bus memory port))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:00
-	67	((encrypt$4 algorithm$2) with (hardware apparatus design system circuit) with (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system)) and ((double dual) adj (bus memory port))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:01
-	68	((encrypt$4 algorithm$2) with (hardware apparatus design system circuit) with (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:28
-	41	((encrypt$4 algorithm$2) with (hardware apparatus design system circuit) with (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:29
-	66	((encrypt$4 algorithm$2 key cipher) with (hardware apparatus design system circuit) with (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:30
-	63	((encrypt$4 algorithm$2 key cipher) same (hardware apparatus design system circuit) same (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher) and ((secur$3 protect$3) with (stor$3 memory))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:33
-	56	((encrypt$4 algorithm$2 key cipher) same (hardware apparatus design system circuit) same (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher) and ((secur$3 protect$3) with (stor$3 memory)) and (programmable fpga gate logic)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:35

-	55	((encrypt$4 algorithm$2 key cipher) same (hardware apparatus design system circuit) same (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher) and ((secur$3 protect$3) with (stor$3 memory)) and (programmable fpga gate logic) and ((exchang$4 transfer$4) with (host computer system))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:46
-	14	((encrypt$4 algorithm$2 key cipher) same (hardware apparatus design system circuit) same (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher) and ((secur$3 protect$3) with (stor$3 memory)) and (programmable fpga gate logic) and ((exchang$4 transfer$4) with (host computer system) with pci)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:37
-	23	((encrypt$4 algorithm$2 key cipher) same (hardware apparatus design system circuit) same (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher) and ((secur$3 protect$3) with (stor$3 memory)) and (programmable fpga gate logic) and ((exchang$4 transfer$4) with (host computer system) with (interface pci))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:43
-	0	(((encrypt$4 algorithm$2 key cipher) same (hardware apparatus design system circuit) same (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher) and ((secur$3 protect$3) with (stor$3 memory)) and (programmable fpga gate logic) and ((exchang$4 transfer$4) with (host computer system) with (interface pci))) not (((encrypt$4 algorithm$2 key cipher) same (hardware apparatus design system circuit) same (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher) and ((secur$3 protect$3) with (stor$3 memory)) and (programmable fpga gate logic) and ((exchang$4 transfer$4) with (host computer system))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:43

-	32	(((encrypt$4 algorithm$2 key cipher) same (hardware apparatus design system circuit) same (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher) and ((secur$3 protect$3) with (stor$3 memory)) and (programmable fpga gate logic) and ((exchang$4 transfer$4) with (host computer system))) not (((encrypt$4 algorithm$2 key cipher) same (hardware apparatus design system circuit) same (parallel simultaneous$3 ("same time"))) and ((isolat$4 or separat$4) with (process$4 system host computer)) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher) and ((secur$3 protect$3) with (stor$3 memory)) and (programmable fpga gate logic) and ((exchang$4 transfer$4) with (host computer system) with (interface pci)))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:43
-	73	((encrypt$4 algorithm$2 key cipher) same (hardware apparatus design system circuit) same (parallel simultaneous$3 ("same time"))) and ((double dual) adj (bus memory port)) and (symmetr$2 asymmetr$2 key secret cipher) and ((secur$3 protect$3) with (stor$3 memory)) and (programmable fpga gate logic) and ((exchang$4 transfer$4) with (host computer system))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 16:48
-	75	"5682027" "5805712"	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 17:49
-	36	5805712.URPN.	USPAT	2004/04/27 17:45

-	75	(US-6427911-$ or US-6431443-$ or US-6453397-$ or US-6484946-$ or US-6488211-$ or US-6505193-$ or US-6507904-$ or US-6510983-$ or US-6542610-$ or US-6633981-$ or US-6643374-$ or US-6647494-$ or US-6659354-$ or US-6678825-$ or US-6687721-$ or US-6697489-$ or US-6700902-$ or US-6641045-$ or US-6629244-$ or US-6633963-$ or US-6401208-$ or US-6389537-$ or US-6425522-$ or US-6427909-$ or US-6427910-$ or US-6571335-$).did. or (US-6575372-$ or US-6603857-$ or US-6389533-$ or US-6357665-$ or US-6385723-$ or US-6357004-$ or US-6202150-$ or US-6209098-$ or US-6219423-$ or US-6220510-$ or US-6230267-$ or US-6233685-$ or US-6260172-$ or US-6293470-$ or US-6297789-$ or US-6317832-$ or US-6320964-$ or US-6328217-$ or US-6354489-$ or US-6125452-$ or US-6145739-$ or US-6151677-$ or US-6151678-$ or US-6164549-$ or US-6167551-$ or US-6181803-$ or US-6189787-$).did. or (US-6014442-$ or US-6021201-$ or US-6052784-$ or US-6058478-$ or US-6070148-$ or US-6076162-$ or US-6088450-$ or US-6091817-$ or US-6095412-$ or US-6115816-$ or US-6122625-$ or US-5905245-$ or US-5912453-$ or US-5923018-$ or US-5949881-$ or US-5974143-$ or US-6000608-$ or US-5682027-$ or US-5805712-$).did. or (US-6145739-$ or US-5682027-$ or US-5805712-$).did.	USPAT; DERWENT	2004/04/27 17:49

-	9628	((US-6427911-$ or US-6431443-$ or US-6453397-$ or US-6484946-$ or US-6488211-$ or US-6505193-$ or US-6507904-$ or US-6510983-$ or US-6542610-$ or US-6633981-$ or US-6643374-$ or US-6647494-$ or US-6659354-$ or US-6678825-$ or US-6687721-$ or US-6697489-$ or US-6700902-$ or US-6641045-$ or US-6629244-$ or US-6633963-$ or US-6401208-$ or US-6389537-$ or US-6425522-$ or US-6427909-$ or US-6427910-$ or US-6571335-$).did. or (US-6575372-$ or US-6603857-$ or US-6389533-$ or US-6357665-$ or US-6385723-$ or US-6357004-$ or US-6202150-$ or US-6209098-$ or US-6219423-$ or US-6220510-$ or US-6230267-$ or US-6233685-$ or US-6260172-$ or US-6293470-$ or US-6297789-$ or US-6317832-$ or US-6320964-$ or US-6328217-$ or US-6354489-$ or US-6125452-$ or US-6145739-$ or US-6151677-$ or US-6151678-$ or US-6164549-$ or US-6167551-$ or US-6181803-$ or US-6189787-$).did. or (US-6014442-$ or US-6021201-$ or US-6052784-$ or US-6058478-$ or US-6070148-$ or US-6076162-$ or US-6088450-$ or US-6091817-$ or US-6095412-$ or US-6115816-$ or US-6122625-$ or US-5905245-$ or US-5912453-$ or US-5923018-$ or US-5949881-$ or US-5974143-$ or US-6000608-$ or US-5682027-$ or US-5805712-$).did. or (US-6145739-$ or US-5682027-$ or US-5805712-$).did.) and (symmetr$3 asymmetr$4) (public and secret)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 17:50

-	38	((US-6427911-$ or US-6431443-$ or US-6453397-$ or US-6484946-$ or US-6488211-$ or US-6505193-$ or US-6507904-$ or US-6510983-$ or US-6542610-$ or US-6633981-$ or US-6643374-$ or US-6647494-$ or US-6659354-$ or US-6678825-$ or US-6687721-$ or US-6697489-$ or US-6700902-$ or US-6641045-$ or US-6629244-$ or US-6633963-$ or US-6401208-$ or US-6389537-$ or US-6425522-$ or US-6427909-$ or US-6427910-$ or US-6571335-$).did. or (US-6575372-$ or US-6603857-$ or US-6389533-$ or US-6357665-$ or US-6385723-$ or US-6357004-$ or US-6202150-$ or US-6209098-$ or US-6219423-$ or US-6220510-$ or US-6230267-$ or US-6233685-$ or US-6260172-$ or US-6293470-$ or US-6297789-$ or US-6317832-$ or US-6320964-$ or US-6328217-$ or US-6354489-$ or US-6125452-$ or US-6145739-$ or US-6151677-$ or US-6151678-$ or US-6164549-$ or US-6167551-$ or US-6181803-$ or US-6189787-$).did. or (US-6014442-$ or US-6021201-$ or US-6052784-$ or US-6058478-$ or US-6070148-$ or US-6076162-$ or US-6088450-$ or US-6091817-$ or US-6095412-$ or US-6115816-$ or US-6122625-$ or US-5905245-$ or US-5912453-$ or US-5923018-$ or US-5949881-$ or US-5974143-$ or US-6000608-$ or US-5682027-$ or US-5805712-$).did. or (US-6145739-$ or US-5682027-$ or US-5805712-$).did.) and ((symmetr$3 asymmetr$4) (public and secret))	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/27 17:50
-	0	6021201.pn. and cmos	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 11:19
-	1	6021201.pn. and serial	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 11:44
-	1	6021201.pn. and (fpga gate or array)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 13:16
-	0	6021201.pn. and eeprom	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 13:16
-	0	6021201.pn. and flash	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 13:16
-	0	6021201.pn. and nonvolatile	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 13:19

-	661	(((flash adj memory) or eeprom or prom) same (bus interface) same (sram or dram)) and (crypto$ or cipher encrypt$3)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 13:24
-	184	(((flash adj memory) or eeprom or prom) same (bus interface) same (sram or dram)) and (crypto$ or cipher encrypt$3) and dma	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 13:25
-	62	(((flash adj memory) or eeprom or prom) same (bus interface) same (sram or dram)) and (crypto$ or cipher encrypt$3) and dma and cmos	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 13:46
-	221	(((flash adj memory) or eeprom or prom) same (bus interface) same (sram or dram)) and (crypto$ or cipher encrypt$3) and cmos	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 13:50
-	37	((((flash adj memory) or eeprom or prom) same (bus interface) same (sram or dram)) and (crypto$ or cipher encrypt$3) and cmos) and intel	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 13:47
-	482	cmos with key	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 16:33
-	81	cmos with key with (contain$3 stor$3 sav$3)	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 16:42
-	4	(cmos with key with (contain$3 stor$3 sav$3)) same reset$4	USPAT; US-PGPUB; EPO; JPO; DERWENT; IBM_TDB	2004/04/29 16:43

TDB-ACC-NO: NB910934

DISCLOSURE TITLE: Cryptographic Microcode Loading Controller for Secure Function.

PUBLICATION-DATA: IBM Technical Disclosure Bulletin, September 1991, US

VOLUME NUMBER: 34

ISSUE NUMBER: 4B

PAGE NUMBER: 34 - 36

PUBLICATION-DATE: September 1, 1991 (19910901)

CROSS REFERENCE: 0018-8689-34-4B-34

DISCLOSURE TEXT:

- This article describes a method of implementing a single-chip microcontroller such that microcode may be loaded, when desirable, in a secure way.
- The ability to load microcode into a microcontroller at IPL is advantageous in that one device could be used in a large number of applications, reducing inventory and cost. Microcode updates could be installed without having to change physical hardware. This also has inventory and cost-saving implications.
- An obstacle to this approach, however, is that many manufacturers of microcontrollers consider the microcode proprietary and do not wish to have it exposed in plain text while awaiting loading. For example, the microcode which implements a disk or LAN controller on a common microcontroller is valuable. If the microcode is made public by ex posing it in plain text in open

storage or on disk, it could simply be copied and used elsewhere. This article describes a method by which microcode could be kept in open storage, protected by encryption, and then loaded into the microcontroller at IPL, where it is decrypted for use.

- In a typical microcontroller, a microprocessor executes microcode out of a Read-Only Memory (ROM), has some data storage (RAM), and reads and writes from some I/O (which is how it connects to the device(s) that it is controlling). The important part, for this discussion, is the microcode ROM. In typical applications the microcode ROM is masked into the microcontroller during fabrication. This effectively turns a general-purpose device into a special-purpose device. At the point where the microcode becomes fabricated into the part, the microcontroller becomes a DASD controller or a LAN adapter, etc., and can no longer be changed or upgraded.

- One way of fixing the problem is to use loadable microcode. This permits a general-purpose device to be customized or updated by writing new or additional microcode. The problem with this, as was mentioned previously, is one of security. If the microcode is exposed in a public area of the system in which the controller is used, it can be copied or changed without authorization. One solution is to encrypt the microcode for transportation and storage, and decrypt it only within the confines of the microcontroller itself.

- Fig. 1 shows a block diagram of this type of system. The microprocessor, I/O, and RAM remain unchanged. The microcode ROM is now split into two segments (both ROM and RAM or EEPROM), and a new storage element (key storage) is defined. The microcode ROM is now a smaller ROM which would be common to all devices. This smaller ROM would have bootstrap and decryption microcode, and be used for initializing and IPLing the microcontroller.

- On power-up, reset, or specific command, this common part of the ROM would load encrypted microcode into a microcode RAM or EEPROM, decrypt it, and begin execution. The microcode could be loaded on existing I/O lines or use a dedicated load path if desired.

To allow the decryption of the microcode, the microcontroller must have the decryption key for the code. This would be kept in the key storage element (RAM or EEPROM).

- Both the key storage and the microcode RAM or EEPROM (once it has been loaded) contain valuable information (the microcode) or the means to get it (the decryption key), so additional security measures should be applied. The microcontroller should be fabricated on a single chip. This prevents probing inter-chip connections. The chip on which the microcontroller is fabricated should be securely encapsulated during packaging. Secure encapsulation is a process by which the chip is coated with a material which obscures the surface of the chip such that it cannot be effectively examined or probed, and if removal is attempted, the chip will be ruined, preventing access to secure data. To provide an additional level of security, the key store and the microcode store could be CMOS RAM. This would make probing or examination more difficult because of

CMOS's sensitivity to light and static charge. These RAMs could be backed with a battery when the system was unpowered (in the same manner as any CMOS RAM). This is a common and inexpensive technique.

- The result would be a general-purpose interface/microcontroller which could securely load microcode for which it had the decryption key. Some of the advantages of such a system are:

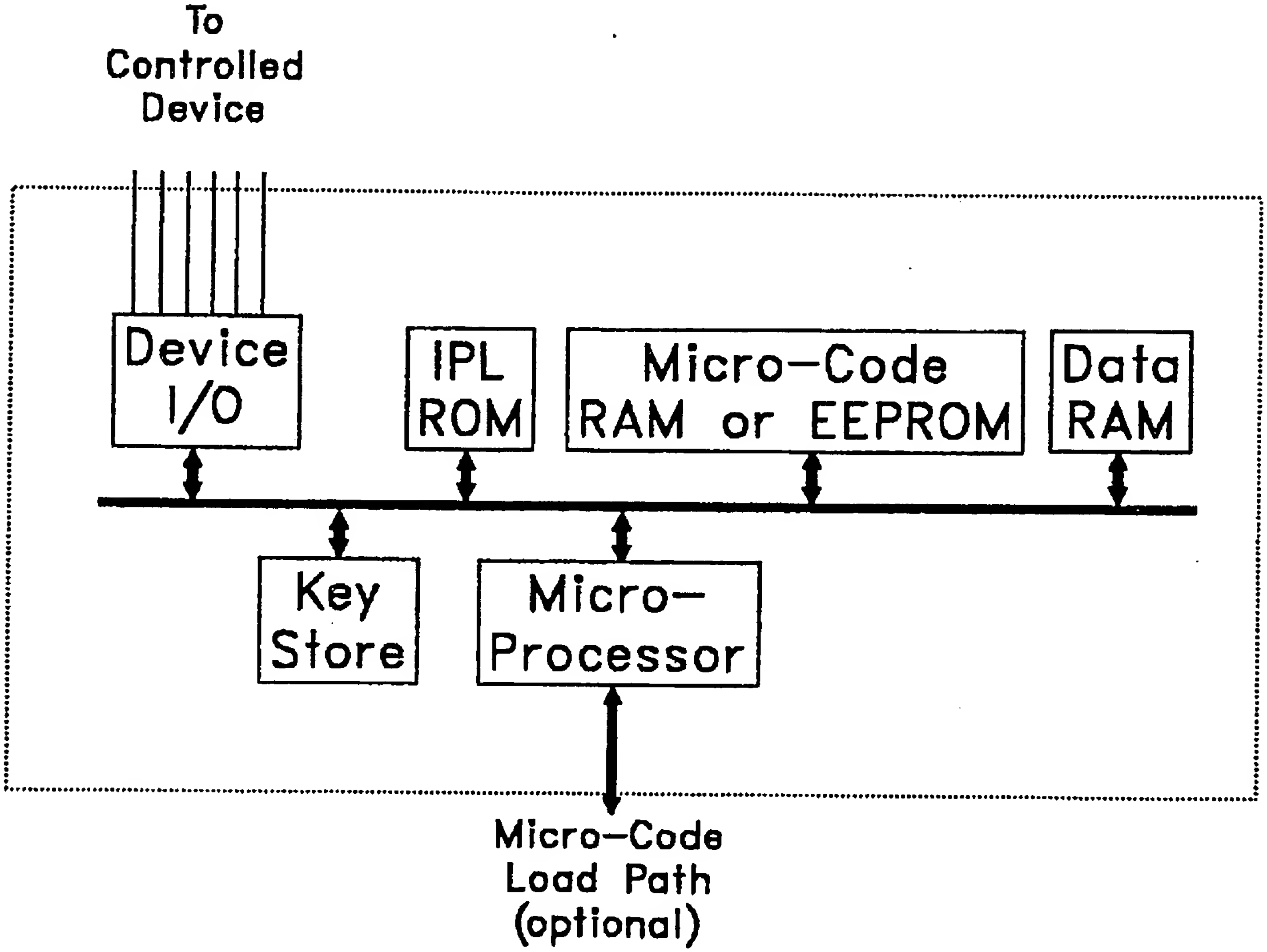
1. A common device could replace a number of customized devices. This is desirable from an inventory and stocking perspective.

- 2. Hardware devices could be field updated in microcode, without replacing hardware.

- 3. Microcode would become loadable and replaceable, without becoming vulnerable.

- 4. Microcode upgrades could be sold for added value.

- The subject of key distribution and management is quite complicated and has not been dealt with here. There are, however, defined methods of key distribution which are appropriate for this type of system.



Cryptographic Micro-controller Block Diagram